NOVEL APPLICATIONS OF OPTIMIZATION TO MOLECULE DESIGN

J.C. MEZA*, T.D. PLANTENGA† AND R.S. JUDSON‡

Abstract. We present results from the application of two conformational search methods: genetic algorithms (GA) and parallel direct search methods for finding all of the low energy conformations of a molecule that are within a certain energy of the global minimum. Genetic algorithms are in a class of biologically motivated optimization methods that evolve a population of individuals where individuals who are more "fit" have a higher probability of surviving into subsequent generations. The parallel direct search method (PDS) is a type of pattern search method that uses an adaptive grid to search for minima. In addition, we present a technique for performing energy minimization based on using a constrained optimization method.

Key Words. global optimization, constrained optimization, nonlinear programming, molecular conformation.

1. Introduction. An important goal of computational chemistry research is the design of molecules for specific applications. Examples of these types of applications occur in the development of enzymes for the removal of toxic wastes, the development of new catalysts for material processing, biosensor design and the design of new anticancer agents. Factors that must be taken into account include shape, size, electronic properties, and reactivity. For many physical and biological properties, the molecular conformation largely determines the final function, and this is the rationale for the development of a large number of conformation search methods.

The general approach is to search the conformation space of a molecule in order to find all energy minima within a prescribed energy range. The problem can be broken into two major parts: defining the energy function and finding efficient methods for performing the conformational search. In general, one can decompose the search into two phases. In the first phase, we are interested in performing a coarse but broad search. This stage generates a number of interesting conformations that can be used as starting guesses for the second phase, which is local energy minimization. The global search phase is conceptually the harder of the two because the size of the parameter space is so large. Additionally, local information about the surface rarely provides definitive clues regarding the location of the global minimum. Because it is difficult to exhaustively search the conformation space of any but the smallest molecules, a number of statistical heuristic methods have been developed [21, 43]. These include pure random search, simulated annealing [45], Cartesian coordinate directed tweak [42], taboo search [11], parallel stochastic methods as in [7, 8], genetic algorithms [23, 24, 27] and direct search methods [29, 32, 33, 46]. Non-stochastic methods have been developed, including Scheraga's diffusion method [25] and a class of branch-and-bound methods due to Floudas [30]. A discussion of many of the methods used for molecular conformation and protein folding can also be found in [37, 38]. All of these methods can be made to work well on a selected set of molecules, but it is important to perform head-to-head tests between different methods to assess their relative strengths.

* Scientific Computing Department, MS 9214, Sandia National Laboratories, Livermore, CA 94551-0969, meza@ca.sandia.gov, supported in part by the Department of Energy under contract DE-ACO4-94AL85000

† tdplant@ca.sandia.gov

‡ rsjuds@ca.sandia.gov

We are primarily concerned here with the efficient broad search of conformation space to generate all of the low energy conformations within a prescribed energy of a global minimum. In this sense, this work is similar in spirit to that of Saunders et al. [43]. In particular, we present a comparison of two search methods, the GA and PDS methods, for this problem. Genetic algorithms draw on a set of evolutionary metaphors including selection of fit individuals, mutation, and genetic crossover. PDS methods belong to a class of optimization algorithms developed by Dennis and Torczon [12] that can be viewed as multidirectional line search methods. These methods are robust, simple to implement, and easily parallelized. We note that neither of these methods has as its main goal that of finding all minima within a certain distance of the global minimum, but earlier work has indicated that both of these methods might be applicable to this problem.

In addition, we present a technique for computing local energy minimization based on a constrained optimization method. This method (originally described in [40]) is based on transforming an unconstrained optimization problem in torsion space into a constrained optimization problem using distance constraints that makes the energy minimization more tractable.

The paper is organized as follows. Section 2 describes the energy functional that we seek to minimize. Section 3 gives an outline of two conformational search methods used in this paper. Section 4 gives a description of the new constrained optimization method for performing energy minimization. In Section 5 we describe the test problems and give numerical results. Section 6 follows with an analysis of the results.

2. Potential Energy Equations. The conformational search problem can be stated as a problem of finding the molecular conformation that yields the lowest energy for a particular N-atom molecule, that is,

$$\min \quad E(x, y, z), \tag{1}$$

where E is the energy of the molecule given its coordinates $x, y, z \in \mathbf{R}^{\mathrm{N}}$. Using this formulation, the conformational search problem can be viewed as a global optimization problem. Unfortunately, because the total energy of a molecule depends on all atom-atom interactions, the number of possible low-energy configurations can grow exponentially with the number of atoms and has been estimated by Hoare to be on the order of $O(e^{N^2})$ for an N-atom molecule [19].

The energy that we wish to minimize can take many forms, but it is usually computed as a sum of terms that are functions of bond distances between two atoms, bond angles between three atoms, dihedral or torsion angles between four atoms, improper torsion angles, various non-bonded terms (Coulombic potentials, van der Waals potentials) and perhaps including solvent effects.

In this paper we will use the potential energy equations used in CCEMD (Center for Computational Engineering Molecular Dynamics) [22]. These equations correspond to the force field used in QUANTA/CHARMM19 [5, 41] with modifications to comply with the force field in CHARMM22. Here we only present a brief statement of the major terms. The total energy is given by

$$E = E_b + E_\theta + E_\phi + E_\omega + E_{LJ} + E_{el}. \tag{2}$$

The first four terms correspond to covalent bond interaction terms between atoms, while the last two terms correspond to nonbonded forces between any two atoms. The bonded energy term is defined by

$$E_b = \sum_{i=1}^{N_{bond}} k_b^i (r^i - r_0^i)^2, \tag{3}$$

where k_b^i is the force constant for bond i, r^i is the bond distance, and r_0^i is the equilibrium bond distance. The bond angle energy term is given by

$$E_\theta = \sum_{i=1}^{N_{angle}} k_\theta^i (\theta^i - \theta_0^i)^2, \tag{4}$$

where k_θ^i is the force constant for bond angle i, θ^i is the bond angle, and θ_0^i is the equilibrium bond angle. The dihedral angle energy term is

$$E_\phi = \sum_{i=1}^{N_{dihedral}} \sum_j |k_i^j| - k_i^j \cos(n_i^j \phi^i), \tag{5}$$

where k_i^j is a force constant, n_i^j is an integer that can take on the values $1, 2, 3, 4, 6$ and ϕ^i is the dihedral angle. The improper torsion angle is defined as

$$E_\omega = \sum_{i=1}^{N_{improper}} k_\omega^i (\omega^i - \omega_0^i)^2, \tag{6}$$

where k_ω^i is the force constant for improper torsion, ω^i is the torsion, and ω_0^i is the equilibrium torsion angle.

In addition to the bonded interactions, there are forces due to non-bonded interactions. The van der Waals term is usually taken to be of the form of a Lennard-Jones potential,

$$E_{LJ} = \sum_{i \neq j} \left(\frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} \right) \quad \mathrm{sw}(r_{ij}), \tag{7}$$

where $A_{ij} = \varepsilon_{ij} \sigma_{ij}^{12}, B_{ij} = 2\varepsilon_{ij} \sigma_{ij}^6$, and $\varepsilon_{ij}, \sigma_{ij}$ depend on the atoms i and j. The term $\mathrm{sw}(r_{ij})$ is a switching function that is used to cutoff the potential at long distances. A variety of switching functions may be used depending on the application (for details see [22]).

The final term due to the electrostatic potential is given by

$$E_{el} = \sum \frac{q_i q_j}{4\pi\varepsilon_0 r_{ij}} \quad \mathrm{sw}(r_{ij}), \tag{8}$$

where q_i and q_j are the charges on atoms i and j respectively and ε_0 is a dielectric constant.

3. Search Methods. In this paper, we make the distinction between a *search* method and a *minimization* method for the conformation problem. By a search method we will mean any algorithm that is used to perform a coarse search of the parameter space to look for starting guesses for a gradient-based local minimization algorithm. In this sense, both genetic algorithms (GA) and parallel direct search (PDS) methods are good candidates for search methods since they can be used to quickly sample a large region of the parameter space. In addition, both of these methods are easily parallelized. This section describes the major features of the GA and PDS methods and how they are applied to the conformation problem.

3.1. Genetic Algorithms. We present here only a brief introduction to our variant of the standard GA method [17]. The most important idea is that we work with a population of *individuals* that will interact through genetic operators to carry out an optimization process. An individual is specified by a *chromosome* that is a bit string of length N_c that can be decoded to give a set of physical parameters. In what follows, chromosome and bit string are synonymous. The function to be optimized, also called the fitness function, is used to rank the individual chromosomes. Optimization proceeds by generating populations whose individuals have increasingly higher fitness. An initial population of N_{pop} individuals is formed by choosing N_{pop} bit strings at random and evaluating each individual's fitness.

Conformations are represented as bit strings that code for the free torsion angles in the molecule. All bond distances and angles are held fixed. Each torsion is represented by n bits giving a resolution of $360/(2^n - 1)$ degrees. Typical values of n range from 6 to 12. If b is the Gray coded binary value of the angle, the value in degrees is $360b/(2^n - 1)$. The chromosome for the individual is constructed by concatenating the bit strings for the individual torsions. The three principal operators are selection of parents, crossover, and mutation.

In our selection operator, every individual in the top ranked 40% of the population has an equal chance of being selected for mating. All individuals in the bottom ranked 60% of the population are discarded. The fitness is the negative of the potential energy. The crossover operator takes a pair of parents and chooses a random cut point along the bit string. The chromosome of the first child is filled in with the bits to the left of the cut point from the first parent and the bits to the right of the cut point from the second parent. The second child gets the complementary bits from the two parents. Note that the crossover point is not restricted to lie on the boundaries separating dihedrals. Doing so restricts the search too much and leads to poorer performance by the GA. Notice however, that this introduces a subtle type of mutation because the dihedral that is disrupted by the cut point does not assume the angular value of either of the parents. By not restricting the cut point positions, we find that premature convergence occurs less often and that lower energies are found.

Finally the mutation operator acts by flipping bits in the binary chromosome. Each bit has a probability equal to R_m of being flipped from 1 to 0 or vice versa. Mutation rates are typically quite low, on the order of 0.04. An important detail is that the entire population is not regenerated at each generation. The top 10% of the old population is moved into the new population and all but the single best are subjected to the mutation operator, meaning that they are mutated with the same low probability as the rest of the members of the new population. We always use the *elitist* strategy in which the most fit individual in each generation is passed directly to the next without crossover or mutation. This ensures that the best individual is never lost, but continues to be available for mating. Note that this individual is transferred

directly from generation i to $i+1$ but also produces offspring that make up part of generation $i+1$.

Additionally, during replication there is a small probability of a bit flip or mutation in a chromosome. This serves primarily to maintain diversity and prevent premature convergence that occurs when a single very fit individual takes over the entire population early in the evolutionary process. To bound the magnitude of the effect of mutations, the binary chromosomes are Gray coded [17]. An integer that is represented as a Gray coded binary number has the property that most single bit flips change the value of the integer by ± 1.

We have the ability to run multiple sub-populations simultaneously. At periodic intervals, these populations can communicate by passing the best individual from each population to each of the others.

During the crossover operations, a niching operation is used. As prospective new members of the population are created, they are compared to those already accepted, by measuring the Hamming distance. The Hamming distance is the fraction of bit positions that have different values in the two chromosomes. The prospective new member is rejected if it is too similar to ones already present. Initially, an individual must differ by 40% from every other individual (that is, no more than 60% of the bits in the two can by set the same.) As the population fills up, this criteria becomes too restrictive and it is slowly relaxed until the population is filled.

3.2. Direct Search Methods. Direct search methods belong to a class of optimization methods that do not compute derivatives. Examples of direct search methods are the Nelder-Mead Simplex method [35], Hooke and Jeeves' pattern search [20], the box method [4], and Dennis and Torczon's parallel direct search algorithm (PDS) [12].

The PDS algorithm can be described as follows. Starting from an initial simplex S_o, the function value at each of the vertices in S_o is computed and the vertex corresponding to the lowest function value, v_o, is determined. Using the underlying grid structure, the simplex S_o is rotated 180° about v_o and the function values at the vertices of this rotation simplex, S_r, are compared against v_o. If one of the vertices in the simplex S_r has a function value less than the function value corresponding to v_o, then an expansion step to form a new simplex, S_e, is attempted in which the size of S_r is expanded by some multiple, usually 2. The function values at the vertices of S_e are compared against the lowest function value found in S_r. If a lower function value is encountered, then S_e is accepted as the starting simplex for the next iteration; otherwise S_r is accepted for the next iteration. If no function value lower than the one corresponding to v_o is found in S_r, then a contraction simplex is created by reducing the size of S_o by some multiple, usually $1/2$ and is accepted for the next iteration.

Because PDS only uses function comparisons, it is easy to implement and use. Since the rotation, expansion, and contraction steps are all well-determined it is possible to determine ahead of time a set of grid points corresponding to the vertices of the simplices constructed from various combinations of rotations, expansions, and contractions. Given this set of grid points, called a search scheme, the PDS algorithm can compute the function values at all of these vertices in parallel and take the vertex corresponding to the lowest function value. An interesting consequence of this approach is that the PDS algorithm can jump out of local wells by using a large enough search scheme size. By varying the size of the search scheme one can therefore use the PDS algorithm as a means of efficiently generating conformations in a manner similar to GA and simulated annealing.

It is also worthwhile to contrast PDS with grid search methods. In a grid search

method the grids are generated by starting with a fixed molecule and systematically varying one of the internal variables. This method works well for small molecules but becomes computationally prohibitive for larger molecules. The grid in PDS however is adaptive and will automatically change in response to the contours of the energy surface.

4. Energy Minimization Using Constraints. The search method chooses candidate conformations for which the fitness must be evaluated. Fitness is usually some measure of the smallest potential energy that can be achieved from the candidate conformation. A simple measure is to compute the energy E for the conformation using equations (1)–(8). Much better results can be obtained by performing a local gradient minimization of the energy starting from the conformation chosen by the search method. However, gradient minimizations can be computationally expensive, especially for large molecules. We have chosen an alternative that makes a physically intuitive compromise. In this approach, the covalent structure of the molecule is fixed and local minimization is performed with only the dihedral angles as variables. This is also known as minimizing in torsion space [34]. The number of dihedral variables is much smaller than the number of atoms, so the minimization problem is easier to solve. In addition, fixing bond distances and bond angles eliminates many local energy minima that are similar in depth; hence, torsion space minima produce a simpler but still physically meaningful picture of the accessible conformations of a molecule.

It is possible to perform energy minimization in torsion space by expressing the potential energy and its analytic derivatives in terms of the dihedral variables. This approach was pioneered by Scheraga and coworkers in their ECEPP program [34] and has also been pursued by Gō et al. [2, 44] and Abagyan et al. [31, 1]. One of the difficulties with performing calculations in torsion space is that a complicated transformation of variables is required to go from Cartesian coordinates to a set of dihedral variables. The CHARMm potential energy model is most naturally expressed as a function of the Cartesian coordinates of each atom. Transforming this model to use dihedral variables and then computing analytic derivatives is quite complicated, usually requiring topological analysis of the molecule, definition of multiple local coordinate frames, and the use of matrix operators to link the local coordinate systems (see [16, 2, 31] for examples). In addition, the number of operations necessary to make a transformation is proportional to the square of the number of dihedral variables; thus, computational costs increase rapidly as larger molecules are examined.

Instead of transforming the potential energy to torsion space, we find a set of distance constraints between pairs of atoms that serves to restrict all molecular motion except rotation of specified dihedral angles. The distance constraints are simple quadratic functions in Cartesian coordinates. Thus, we minimize the usual potential energy function subject to a set of quadratic equality constraints, all in Cartesian coordinates. Our constrained problem is equivalent to minimizing in torsion space in the sense that we find the same set of local minima. However, the intermediate molecular conformations generated during our minimization are different. Our approach avoids the mathematical complexities associated with transforming to dihedral variables. We have a simple method of finding an appropriate set of distance constraints that is easily automated. Also, we are able to maintain variable sparsity in our constrained formulation, keeping the linear algebra costs manageable even for large molecules.

The chief advantage of our approach is that it makes numerical energy minimization in torsion space much simpler. Our method poses a minimization problem with nonlinear equality constraints in Cartesian coordinates instead of an unconstrained

problem in internal coordinates. We have employed a code [26] based on sequential quadratic programming (SQP) methods that seems well suited to the constrained molecular mechanics problem. An SQP method treats the constraints explicitly and does not require the elimination of the dependent variables from the nonlinear constraints. The algorithm utilizes sparse linear algebra techniques to solve all subproblems and operates as a quasi-Newton or truncated Newton method with only first derivative information.

4.1. Dihedral variables and constraints. We follow Scheraga [34] and fix the covalent structure of the molecule except for the rotation of dihedral angles. This simplified model determines an approximate molecular conformation by changing dihedrals in response to non-bonded and dihedral angle forces. Dihedral angles are now the primary variables of the problem – atom positions are computed from knowledge of the dihedrals and the fixed bond distances and angles. An unconstrained minimization can be carried out in torsion space if we can calculate the gradient of the potential energy with respect to dihedral variables; that is, if we can transform the Cartesian force vector into dihedral coordinates. The transformation can be done analytically as in [2, 31], but the equations are extremely complicated to derive and to program. We wish to show how the transformation of coordinates can be avoided using distance constraints between pairs of atoms.

Let us consider the simple example of ethane in Figure 1. It has a single dihedral angle, whose rotation is illustrated by the arrow. Usually, two planes are specified, such as H1-C1-C2 and C1-C2-H2, and the dihedral is defined as the angle between the two planes about the axis C1-C2. Now if all bond distances and angles in the ethane molecule are fixed, then the methane group on the left containing C1 and H1 is a rigid body that rotates about the C1-C2 axis. The methane group on the right is a similar rigid body. Thus, rotation about a dihedral can be characterized as the relative rotation of two rigid bodies about a common axis.

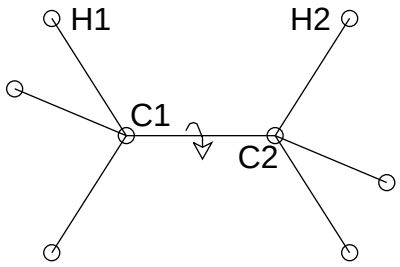

FIG. 1. *Dihedral rotational in ethane*

Figure 2 shows the ethane molecule with distance constraints (drawn as dashed and dotted lines) for the two methane groups. The distance constraints can be thought of as rigid bars or virtual bonds between atoms that restrict motion. The dashed constraints define a rigid five-point polyhedron on the left and the dotted constraints do the same on the right. The figure shows that these two rigid pieces are connected together in such a way that they can spin about the C1-C2 axis, but cannot otherwise move in relation to each other. Note in particular that the distance constraints for each rigid piece include the dihedral axis and reach across it.

In Figure 2 there are 8 dashed, 8 dotted, and one dash-dot line between C1 and C2, for a total of 17 constraints. In general a set of n points describing a rigid body

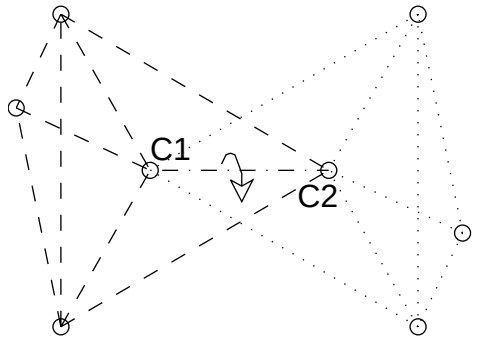

FIG. 2. *Distance constraints allowing only dihedral rotation in ethane. There are 17 constraints restricting molecular motion.*

in three dimensions has six degrees of freedom (three for translation and three for rotation with respect to an external coordinate system); therefore, $3n-6$ constraints are needed to make the body rigid. Each rigid polyhedron in Figure 2 contains 5 atoms and $(3\times 5)-6=9$ distance constraints (the C1-C2 constraint is shared by both). The ethane molecule as a whole can freely translate and rotate, and its dihedral angle gives it one internal degree of freedom; therefore, it should have $(3\times 8)-6-1=17$ constraints. This is exactly what Figure 2 shows.

Our method generalizes easily to molecules with more than one dihedral. Suppose there are two dihedrals dividing a molecule into three pieces. The idea is to make each piece into a rigid polyhedron, then connect them together in pairs to allow rotation about the two dihedral axes. The end polyhedrons are treated just like the rigid methane groups in ethane. The middle polyhedron connects to two dihedral axes, but these dihedrals are necessarily distinct and do not interact with one another. Clearly we can extend this idea to any number of dihedrals, as long as they are truly free to rotate in the molecule (a single dihedral in a closed chain does not have full freedom, for instance).

Our procedure allows us to choose distance constraints for each rigid piece independently of the other pieces; that is, it is a strictly *local* procedure. Internal coordinate methods also divide the molecule into rigid pieces, but require a topology tree of interdependencies between the pieces [2, 31]. The tree describes which pieces move when a particular dihedral rotates, and then a calculation is made involving all affected pieces to determine the constrained motion resulting from a dihedral rotation. The global interdependence of pieces results because internal coordinate methods are eliminating variables. We add constraints to the problem instead of eliminating variables, side-stepping the problem of calculating coupled rigid body motions.

4.1.1. Choosing the best constraints for a rigid piece. We specified 9 distance constraints to make each of the five-atom polyhedrons in Figure 2 rigid. There are 10 possible atom-to-atom distance pairs, only 4 of which correspond to the length of a real chemical bond. This is not an unusual situation. In general, a set of n points has $(n^2-n)/2$ possible distance pairs, of which we need $3n-6$. A molecule with no rings has only $n-1$ chemical bonds, so we face a growing surplus of constraint options as n increases.

For a rigid piece with $n>4$ atoms we could add on distance constraints atom-by-atom as in [10]. This is a logical procedure, but it turns out that from an optimization

perspective there is a "best" set of $3n-6$ distance constraints. Each constraint is a quadratic equation in six unknown atom positions. For example, if R_{ij} is the fixed distance between atoms i and j, then the corresponding constraint equation is

$$(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2 = R_{ij}^2.$$

To solve the constrained optimization problem we will employ the transpose of the constraint Jacobian matrix defined by the gradients of the constraint equations. It is important to an optimization algorithm that this matrix be numerically well-conditioned; that is, that its columns be as linearly independent as possible. We define the "best" set of distance constraints as the set for which the matrix of gradients has the smallest condition number.

The best set of constraints is found automatically using a rank-revealing QR factorization [18]. We first assemble a matrix containing the gradient vectors for every possible distance constraint, arranged in any order. The rank-revealing factorization is a Gram-Schmidt orthogonalization procedure that chooses the next column (that is, constraint gradient) to be eliminated by examining the size of all remaining pivot elements. It passes through the entire matrix and returns an optimal ordering of constraints with their pivot sizes. The first $3n-6$ constraints chosen by the factorization give a matrix with the desired small condition number. We use the LINPACK [13] subroutine dqrdc to perform the factorization.

A moment's consideration of Figure 2 reveals that we do not have total freedom in choosing our constraint set. We must make sure that we include the distance constraints that fix the lengths of any dihedral axes connected to our rigid piece (for ethane, this is the rigid bar C1-C2). Otherwise, two connecting rigid polyhedrons would have the freedom to shift along their common dihedral axis as well as rotate around it. Fortunately, dqrdc has the capability of forcing specified constraints to be in the front of the QR factorization matrix.

One other important detail needs discussion. What if the atoms forming a rigid section of the molecule are coplanar? In this case the rank-revealing QR factorization finds only $3n-7$ nonzero pivots. The corresponding distance constraints force the atoms to be rigidly connected within a plane, but they do not force the atoms to remain coplanar (see [3, p. 95], for instance). As discussed in [40], subroutine dqrdc can easily detect nearly coplanar sets of atoms, and CCEMD [22] allows us to introduce fictitious noncoplanar atoms that keep the constraint Jacobian well-conditioned.

Let us summarize our method for choosing the distance constraints. A molecule is provided with bond lengths and bond angles already fixed at desired values, and a set of d dihedrals is specified.

Procedure for defining the distance constraints.

1. Using the d dihedral angles, partition the molecule into $d+1$ nonoverlapping pieces, assigning each atom to exactly one piece. (We assume for simplicity that closed rings do not have free dihedrals.)
2. For the ith piece, i running from 1 to $d+1$:

   a. Define a set S_i consisting of all the atoms in the piece, plus the atom on the opposite of every dihedral axis connected to this piece. Let n_i denote the number of atoms in S_i.

   b. Consider all pairwise distances between points in the set S_i, and construct a matrix whose columns are the gradients of these quadratic distance constraint equalities. The matrix has $3n_i$ rows and $(n_i^2 - n_i)/2$ columns.

c. Perform a rank-revealing QR factorization on the matrix to find the best $3n_i - 6$ distance constraints. Force the factorization to include the distance constraints between atoms that define the dihedral axes touching this piece. (If the magnitude of pivot number $3n_i - 6$ is less than 10^{-2} times the magnitude of the first pivot, then add a fictitious noncoplanar atom to S_i and go back to step **2b**).

3. Combine all the distance constraints generated for each piece. Notice that the constraint between the two atoms defining a dihedral axis shows up twice because it was specifically included in each of the two pieces it joins. Keep just one of these two copies.

A straightforward calculation shows that this procedure generates the correct number of constraints. Let n be the total number of atoms in the molecule. The whole molecule has 6 external and d internal degrees of freedom, so this procedure should find a total of $3n-6-d$ distance constraints. Let d_i be the number of dihedral angles touching the ith piece. Since each dihedral touches exactly two different pieces, we see that

$$\sum_{i=1}^{d+1} d_i = 2d.$$

In step **2a** we included in S_i the atom on the opposite side of the axis for each dihedral touching a piece. Taking it back out gives a strict partitioning of the molecule, so

$$\sum_{i=1}^{d+1} (n_i - d_i) = n.$$

Now the total number of constraints collected in step **3** is just the sum of the number found for each piece minus the extraneous copy of each dihedral axis constraint. If we simplify this number using the previous equations, we obtain the correct number of constraints:

$$\begin{aligned} \sum_{i=1}^{d+1} (3n_i - 6) - d &= \sum_{i=1}^{d+1} (3n_i - 3d_i) + \sum_{i=1}^{d+1} (3d_i - 6) - d \\ &= 3n + 3(2d) - 6(d+1) - d \\ &= 3n - 6 - d. \end{aligned}$$

This procedure also works when free dihedrals are present in closed loops of atoms, provided we check that no duplicate distance constraints are generated between the rigid pieces comprising a loop. Duplicate constraints could conceivably arise between pairs of atoms on different dihedral axes within a loop. This is unlikely if the loop contains many dihedrals, as, for example, in our work with proteins where loops result from cysteine disulfide bridges.

4.2. An optimization algorithm for large-scale constrained optimization problems. In the previous section we showed how to define distance constraints so that only specific dihedral variables can change. This section describes the optimization algorithm used to solve the constrained energy minimization problem. The theory underlying this algorithm stems from the work of Byrd [6] and Omojokun [36] in the area of trust regions for equality constrained optimization. A general purpose

software implementation of the algorithm called ETR was created with large-scale applications in mind [26, 39].

We are faced with solving the constrained algebraic optimization problem

$$\text{(9)} \qquad \min \qquad E(x_1, y_1, z_1, \ldots, x_n, y_n, z_n)$$

$$\text{(10)} \quad \text{subject to} \qquad (x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2 = R_{ij}^2, \quad \text{for} \quad i, j \in \mathcal{D},$$

where the position of atom i in a Cartesian coordinate system is denoted by the triple (x_i, y_i, z_i), the fixed Euclidean distance between atoms i and j is the constant R_{ij}, and $\mathcal{D}$ is an index set containing the full list of distance constraints. The variables in this problem are the $3n$ coordinates of the atoms. The number of equality constraints is $3n - 6 - d$, which can be a large number if we specify only a few dihedrals to be free. The potential energy E is calculated in Cartesian coordinates by some molecular dynamics code in accordance with a given force field model. We assume that E is a continuously differentiable function of the variables and that its gradient (the negative of the force on every atom) can be calculated.

4.2.1. ETR algorithm for equality constrained optimization. The ETR algorithm is based on sequential quadratic programming (SQP), a standard approach for solving optimization problems with nonlinear equality constraints [14, 15]. To use more general notation, let $x \in \mathbf{R}^{\mathrm{N}}$ be the vector of variables, $f(x) : \mathbf{R}^{\mathrm{N}} \to \mathbf{R}$ the function to be minimized, and $c(x) : \mathbf{R}^{\mathrm{N}} \to \mathbf{R}^{\mathrm{M}}$ the set of M equality constraints. The general constrained minimization problem is then written as

$$\text{(11)} \qquad \min_x \ f(x) \qquad \text{subject to} \quad c(x) = 0.$$

Basically, an SQP method adds a Lagrange multiplier variable λ_i for each of the M constraints and applies Newton's method to the resulting system of equations. The Newton method generates a sequence of iterates $\{x^0, x^1, \ldots, x^k, \ldots\}$ that converge to a solution of problem (11). For a given iterate x^k, a quadratic Taylor series expansion of (11) determines the SQP subproblem

$$\text{(12)} \qquad \min_p \qquad f(x^k) + p^T \nabla f(x^k) + \tfrac{1}{2} p^T W(x^k, \lambda^k)\, p$$

$$\text{(13)} \quad \text{subject to} \qquad c_i(x^k) + p^T \nabla c_i(x^k) = 0, \qquad \text{for} \ \ i = 1, \ldots, \mathrm{M}.$$

We minimize this simpler subproblem to find the next iterate x^{k+1}. Note that $f(x^k)$, $c(x^k)$, $\nabla f(x^k)$, $\nabla c_i(x^k)$, and $W(x^k, \lambda^k)$ are constant quantities in the optimization subproblem. The variable $p \in \mathbf{R}^{\mathrm{N}}$ is a distance vector from the point x^k, so that (12) represents a quadratic function in the components of p and (13) defines linear approximations to the constraints that p must satisfy. The matrix W is the Hessian of the Lagrangian

$$W(x^k, \lambda^k) = \nabla^2 f(x^k) + \sum_{i=1}^{\mathrm{M}} \lambda_i^k \nabla^2 c_i(x^k),$$

which contains all second-order derivative information. The solution to (12)-(13) is a vector p^k, known as the *step* at the current iterate x^k. In a pure Newton method the next iterate is calculated directly as $x^{k+1} = p^k + x^k$.

For unconstrained optimization it is well-known that Newton's method only works if the starting iterate x^0 is already close to a solution. The same is true for constrained

optimization. To make the SQP method work for an arbitrary starting guess we employ a *trust region* globalization technique. (Another common globalization technique uses line searches.) This is just a mechanism for judging the accuracy of subproblem (12)-(13) as a suitable model for the real problem (11), which is not quadratic. The trust region is a hypersphere about the point x^k with radius Δ. It is used to limit the length of the step p by appending to subproblem (12)-(13) the inequality

$$\|p\|_2 \leq \Delta \tag{14}$$

($\|\cdot\|_2$ stands for the Euclidean norm of a vector). The idea is to make Δ smaller when the accuracy of the model seems poor. Since a significant amount of work goes into constructing and solving the SQP subproblem, however, a smart trust region algorithm acts to increase Δ when the accuracy seems good, thereby allowing bigger steps towards a solution.

There are a variety of ways to enforce the trust region inequality (14) while solving (12)-(13). We use the method of Byrd [6] and Omojokun [36] because it allows efficient solution of large-scale problems. Our software implementation of this method is called ETR (for *E*quality constrained optimization using *T*rust *R*egions) and is explained in detail in [26, 39].

The ETR code computes a step p^k as the sum of two orthogonal vectors. One of these, the *vertical step*, attempts to satisfy the linearized constraint equations (13). If we collect the gradient vectors of each constraint into the $\mathrm{N} \times \mathrm{M}$ matrix

$$A(x^k) = [\nabla c_1(x^k) \;\; \nabla c_2(x^k) \;\; \cdots \;\; \nabla c_{\mathrm{M}}(x^k)], \tag{15}$$

then all M equations in (13) can be written collectively as $[A(x^k)]^T p^k + c(x^k) = 0$. It turns out that the vertical step $v \in \mathbf{R}^N$ computed by ETR always lies in the range space of $A(x^k)$; that is, v is a linear combination of the columns of $A(x^k)$.

The other part of p^k is called the *horizontal step*. It seeks to minimize the function (12) without disturbing the improvements made by v. To accomplish this it must be orthogonal to every constraint gradient, so we use sparse linear algebra techniques to construct an $\mathrm{N} \times (\mathrm{N} - \mathrm{M})$ matrix Z^k that satisfies the equation $[A(x^k)]^T Z^k = 0$. Then the horizontal step is expressed as a vector $Z^k u$, where $u \in \mathbf{R}^{\mathrm{N}-\mathrm{M}}$ are variables that are chosen to minimize (12) as much as possible. The $(\mathrm{N} - \mathrm{M})$-dimensional subspace spanned by the columns of Z^k is the reduced subspace of problem (12)-(13); that is, the subspace left after imposing the M linearized constraints of (13). The vector u has one component corresponding to each degree of freedom in problem (12)-(13).

ETR forms the step as $p^k = v + Z^k u$. To judge the accuracy of this step we use the merit function $f(x^k + p^k) + \mu\|c(x^k + p^k)\|_2$, where $\mu > 0$ is a parameter that controls the relative importance of minimizing f and of satisfying the equality constraints. The method for choosing μ and other important details are documented in [26].

4.2.2. Applying the ETR algorithm. For a molecule with n atoms and d free dihedrals, the size of the optimization problem is $\mathrm{N} = 3n$ and $\mathrm{M} = 3n - 6 - d$. It takes one computation of the potential energy and interatomic forces in Cartesian coordinates to get $f(x^k)$ and $\nabla f(x^k)$. The values of the constraints and their gradients must also be computed to get $c(x^k)$ and $\nabla c_i(x^k)$; however, these are much cheaper (the computation is roughly equivalent to evaluating the energy and forces due to just the bond lengths). The second derivative information in W is usually approximated by a quasi-Newton matrix. A classical BFGS approximation is appropriate, but for

large problems the storage requirements of the full matrix or its representation can become prohibitive. For this reason we use a compact limited memory BFGS [28, 9] approximation for W, which stores only 10 vectors of length $3n$. Thus, the cost of setting up subproblem (12)-(13) is determined primarily by the cost of one evaluation of the potential energy and forces.

The vertical step v depends on $c(x^k)$ and the constraint gradients collected in $A(x^k)$, both of which were computed in setting up the SQP subproblem. ETR calculates v by treating the linearized constraint equations (13) as a linear least squares problem with a trust region; that is, by solving

$$\min_v \ \|c(x^k) + [A(x^k)]^T v\|_2 \qquad \text{subject to} \quad \|v\|_2 \le 0.8\Delta.$$

Computing v involves some linear algebra operations with the matrix $A(x^k)$, but these are fairly cheap because we chose the distance constraints to make $A(x^k)$ well-conditioned, and because this is an extremely sparse matrix. As explained in [40], the sparsity of this matrix is a distinct advantage over methods which transform all Cartesian variables to dihedral variables.

Computation of the horizontal step is similar to a standard unconstrained quasi-Newton minimization of the potential energy. The main differences are the presence of Lagrange multipliers in the quasi-Newton Hessian approximation, and the restriction that the horizontal step be in the form $Z^k u$. But the multipliers and Z^k both derive from A^k, which is well-conditioned and computationally cheap to work with. Also, the complexity of dealing with Z^k is somewhat offset by the smaller size of the reduced space minimization subproblem (its dimension is $\mathrm{N} - \mathrm{M} = 6 + d$).

In summary, we expect the cost of solving each SQP subproblem to be dominated by the cost of evaluating the potential energy and interatomic forces at the current iterate. There is one force evaluation per subproblem, the same as in most unconstrained minimization algorithms. The extra overhead of solving for nonlinear constraints is not large and should scale linearly with the size of the molecule. If we assume that the potential energy and interatomic forces can be calculated separately, then we obtain the simple outline of the ETR algorithm shown below.

General description of the ETR algorithm for solving (9)-(10).

1 Choose a molecular conformation and load initial atom positions into x^0
2 Make one energy evaluation to get $f(x^0)$ and $c(x^0)$
3 Initialize $\Delta > 0$, $W^0 = I$, and $k = 0$
4 Begin the main loop
  Make one force evaluation to get $\nabla f(x^k)$, and compute $A(x^k)$ using (15)
  Compute Z^k from $A(x^k)$
  Use $A(x^k)$ and $\nabla f(x^k)$ to compute Lagrange multiplier estimates λ^k
4a **if** $\|\nabla f(x^k) - A(x^k)\lambda^k\|_\infty < \epsilon$ **and** $\|c(x^k)\|_\infty < \epsilon$ **then return** *success*
4b Use $A(x^k)$ and $c(x^k)$ to compute a vertical step such that $\|v\|_2 \le 0.8\Delta$
  Use $\nabla f(x^k)$, W^k, and Z^k to compute a horizontal step with $\|Z^k u\|_2^2 \le \Delta^2 - \|v\|_2^2$
  Set $p^k = v + Z^k u$
  Make one energy evaluation at the new trial point to get $f(x^k + p^k)$ and $c(x^k + p^k)$
  if the trial point is not a sufficiently good improvement over x^k
    then $\Delta \leftarrow \gamma_1 \|p^k\|_2$, **goto 4b**
    else $x^{k+1} = x^k + p^k$, $\Delta \leftarrow \gamma_2 \Delta$, update the ℓ-BFGS matrix W^k
  Increment k and **goto 4**

The main loop of the ETR algorithm sets up a series of SQP subproblems. The inner loop beginning at **4b** finds a suitable step p^k that solves the subproblem. ETR decides at **4a** that it has converged if first-order optimality conditions are satisfied to a tolerance ϵ. This means that every distance constraint is within ϵ of its proper length, and that every component of the reduced gradient is smaller than ϵ. The trust region size Δ is updated after every trial point using the parameters γ_1 and γ_2. For best algorithm performance we used $0.1 \leq \gamma_1 \leq 0.5$ and $2 \leq \gamma_2 \leq 5$, with the exact value determined by how much improvement was made in the merit function by the trial point (see [26] for details).

5. Computational test results. In this section we present computational results which show the relative effectiveness of two global search strategies. The GA and PDS algorithms were used to generate a large number of candidate starting conformations. The fitness of each candidate was evaluated by calculating a local energy minimum in torsion space using our constrained optimization method. From this data we plotted the low energy spectrum revealed by each global search scheme. We expect to observe that both the GA and PDS algorithms preferentially find low energy minima. Our experiments provide some quantification of the effectiveness of the search strategies. In addition, we will see whether the methods reveal the structure of the spectrum at higher energies.

5.1. Test Problems. We chose two small synthetic peptides for this investigation, whose characteristics are summarized in Table 1. The peptides were prepared by using QUANTA [41] and were built with no hydrogens to reduce the CPU time for the energy calculations. The energy of each molecule was first gradient-minimized without constraints to form the reference conformation. Although the test molecules are fairly small, they possess a large number of distinct local energy minima. Assuming a simple three-fold symmetry about each dihedral, we expect on the order of $3^4 \approx 100$ distinct minima in torsion space for Thr-Ala, and $3^8 \approx 6500$ for Thr-Ala-Leu. Our objective is to identify all the minima within 10 kcal/mol of the global minimum.

TABLE 1
Test molecule characteristics

sequence	*number of atoms*	*number of dihedrals*
Thr-Ala	13	4
Thr-Ala-Leu	21	8

Each global search strategy varied the dihedral angles to generate a particular starting conformation of the molecule. Bond distances and bond angles were held fixed during this procedure. Our constrained optimization method was applied to each starting conformation and run until a local energy minimum in torsion space was found. The constraint equations were enforced so that rigid interatomic distances did not change by more than 10^{-5} Å. We carried out the local minimization to an unusually tight tolerance, requiring the Euclidean norm of the force vector expressed in dihedral variables to be less than 10^{-5} kcal/mol/Å. The tolerance is more accurate than the chemical model warrants, but our goal was to reliably distinguish between neighboring energy minima and provide a complete map of all low energy minima, similar to [43]. The size and average execution time of the constrained minimization problems are reported in Table 2. All calculations were performed on an SGI Power Challenge with a 75 MHz MIPS R8000 processor.

TABLE 2
Constrained optimization problem sizes

problem	*number of unknowns*	*number of constraints*	*average CPU time*
Thr-Ala	39	29	0.62 seconds
Thr-Ala-Leu	63	49	4.15 seconds

5.2. Energy Minimization Results. We accumulated data for three variants of the GA and PDS search strategies, which are listed in Table 3. The three runs differed primarily in the number of candidate conformations that were generated: 1,000 for the first run, 5,000 for the second, and 40,000 for the third run. In addition, we generated start points from a completely random distribution of dihedral angles. We do not suggest that this is a viable search strategy; it was used merely to help fill in the energy spectrum of the test molecules.

The three GA runs differed in the size of the population making up each generation and in the number of generations, as shown in Table 3. The third run also included niching operations between four sub-populations. The 'chromosomes' of the GA runs were 20-bit representations of the dihedral angle variables. We used a mutation rate of 0.01.

The three PDS runs were identical except for the total number of candidate points generated and the search scheme size. The search scheme size was set to 64 vertices for the Thr-Ala problem and to 256 vertices for the Thr-Ala-Leu problem. In addition, we modified the standard PDS algorithm so that it didn't generate any contraction points in the scheme. The effect of this modification is to allow the method to generate a coarser but broader scheme. Since we are using PDS solely as a search method and we are not concerned with finding a local minimum this allows us to explore more points overall.

TABLE 3
Description of global search strategies. Each line shows a search strategy for choosing dihedral angles to generate different molecular conformations.

0	Each dihedral treated as a uniformly random variable
1	GA with 20 generations, 50 individuals per generation
2	GA with 50 generations, 100 individuals per generation
3	GA with 100 generations, 100 individuals per generation, and 4 niches
4	PDS for a total of 1,000 conformations
5	PDS for a total of 5,000 conformations
6	PDS for a total of 40,000 conformations

The set of local energy minima found from each global search scheme was collected and analyzed for unique conformations. This was done by clustering together final conformations whose energies differed by less than 0.00005 kcal/mol and whose dihedral angles differed by less than 0.1 degree rms. This first clustering criteria was applied to filter out "distinct" local minima which we feel were distinct only because the gradient minimization was not carried to a higher precision. Then a second clustering operation was applied to reduce the minima to a more chemically meaningful set. The members of each of these clusters had energies within 0.1 kcal/mol and dihedrals within 1.0 degree rms. To form these clusters, we examined the list of minima from lowest energy to highest and placed each conformation in an existing

cluster if its energy and dihedrals differed by less than the tolerances from every other conformation already in the cluster.

We report the total number of each cluster type found by the different search strategies in Tables 4 and 5. Each line shows results for one of the global search strategies described in Table 3. The first two columns show the total number of starting points considered by each strategy (*# start pts*) and the lowest energy found. The next two pairs of columns each give the number of mathematically distinct local minima (*# math min*) and chemically relevant distinct minima (*# chem min*). The former are separated by at least 0.00005 kcal/mol or 0.1 degree rms and the latter by 0.1 kcal/mol or 1.0 degree rms. These tables also list the number of local minima found within 10 kcal/mol of the "global" minimum under the *Low energy minima* heading. We take as our estimate of the global minimum potential energy the smallest energy found by any method during our calculations.

TABLE 4
Search results for Thr-Ala. Low energy minima have energies < 17.9791 *kcal/mol.*

			All minima		Low energy minima	
strategy	*# start pts*	*lowest energy*	*# math min*	*# chem min*	*# math min*	*# chem min*
0	90,000	7.9791 kcal/mol	166	70	56	20
1	1,000	7.9791 kcal/mol	47	33	24	15
2	5,000	7.9791 kcal/mol	64	36	34	17
3	40,000	7.9791 kcal/mol	95	45	46	20
4	1,000	7.9791 kcal/mol	63	37	27	13
5	5,000	7.9791 kcal/mol	93	48	39	18
6	40,000	7.9791 kcal/mol	137	64	50	21

TABLE 5
Search results for Thr-Ala-Leu. Low energy minima have energies < -15.2766 *kcal/mol.*

			All minima		Low energy minima	
strategy	*# start pts*	*lowest energy*	*# math min*	*# chem min*	*# math min*	*# chem min*
0	90,000	-24.9003 kcal/mol	8270	4253	1530	695
1	1,000	-24.7243 kcal/mol	433	371	181	144
2	5,000	-25.2766 kcal/mol	1190	893	527	329
3	40,000	-25.2766 kcal/mol	3242	2042	1253	623
4	1,000	-24.7243 kcal/mol	281	238	98	78
5	5,000	-24.9920 kcal/mol	1495	1113	404	274
6	40,000	-25.2766 kcal/mol	5108	2964	1188	587

Table 4 shows that the global minimum for the small Thr-Ala molecule was relatively easy to locate. However, the full set of chemically meaningful low energy states was harder to locate. From strategies 2 and 5 we see that searching over 5,000 candidate conformations turned up 80 % of the low energy minima. Strategies 3 and 6 show that up to 40,000 start points were needed to find all the low energy states.

The number of unknowns doubled in the molecule Thr-Ala-Leu and from Table 5 we see that both strategies were successful in finding the global minimum. The GA and PDS algorithms located approximately the same number of low energy minima, but PDS found significantly more high energy states for the same amount of work (compare strategy 2 with 5, and strategy 3 with 6).

We plotted the local energy minima found by each global search strategy in Figures 3 and 4. Each column in these figures shows the energy spectrum found by a

particular search strategy. Each dash represents one local energy minimum after the second clustering operation was applied; that is, the figures correspond to data in the columns of Tables 4 and 5 headed by *# chem min*. In Figure 4, we have plotted all the energy minima, while Figure 5 shows only the minima within 10 kcal/mol of the apparent global minimum.

From Figure 3 we see that both the GA and PDS methods succeeded in mapping the full energy spectrum of the molecule. PDS found a slightly greater density of states among the higher energies for the same amount of work (compare columns 2 and 5 in the figure). The general structure of the spectrum was evident after only 1000 candidate conformations (columns 1 or 4).

Figure 4 plots only the low energy states of the spectrum of Thr-Ala-Leu. It is apparent that a large number of conformations must be examined to fill in the spectrum, especially at the lowest energies. With 5000 or fewer starting points (strategies 2 and 5), GA does a noticeably better job at finding the lower energy conformations than PDS. As the number of starting points increases however, this difference disappears.

6. Summary. We have presented a comparison of two search methods, GA and PDS, for finding all of the local minima within a prescribed distance of the global minimum energy of a molecule. The GA method is an optimization algorithm designed to find the global minimum of a function. The PDS method is a local optimization method that we have employed as a search method. Although neither of the two methods was designed for the purpose of finding more than one minimum, we have shown that in combination with a local gradient-based minimization method they can find a large number of local minima. Both methods tend to concentrate the computed minima towards the lower energies in the energy spectrum as the sample size of starting points is increased. In this sense, it can be argued that both methods would be appropriate for performing conformational searches.

We have also described some recent work [40] that uses distance constraints between atoms to allow potential energy minimization of molecules while holding all bond lengths and bond angles fixed. The constrained energy minima found by this method are identical to those found by minimizing in torsion space. Our method operates directly in Cartesian coordinates and avoids the usual difficulties associated with transforming to internal coordinates. We have presented a simple procedure for choosing appropriate distance constraints based on linear algebra considerations. It is simple because our constraints are determined solely by the atoms in a single rigid piece of the molecule – no analysis of coupled rigid body motions is needed. Our method requires the solution of a constrained optimization problem in $3n$ Cartesian unknowns instead of an unconstrained problem in d dihedral variables. By employing an optimization algorithm that exploits the sparsity structure of the constraints the new method has an added advantage over the apparently smaller minimization problem in internal coordinates.

REFERENCES

[1] R. Abagyan, M. Totrov, and D. Kuznetsov *ICM – a new method for protein modeling and design: applications to docking and structure prediction from the distorted native conformation*, J. Comp. Chem., 15:488–506, 1994.

[2] H. Abe, W. Braun, T. Noguti, and N. Gō *Rapid calculation of first and second derivatives of conformational energy with respect to dihedral angles for proteins: general recurrent equations*, Comp. and Chem., 8:239–247, 1984.

[3] M.P. Allen and D.J. Tildesley *Computer Simulation of Liquids* New York: Oxford UP, 1987.

[4] G. Box and K. Wilson, *On the experimental attainment of optimum conditions*, J. Royal Statistical Society, Series B, 13 (1951), pp. 1–45.

[5] B.R. Brooks, R.E. Bruccoleri, B.D. Olafson, D.J. States, S. Swaminatha, and M. Karplus, *CHARMM: a program for macromolecular energy, minimization, and dynamics calculations*, J. Comp. Chem., 4:187–217, 1983.

[6] R.H. Byrd *Robust trust region methods for constrained optimization*, Third SIAM Conference on Optimization. Houston, 20 May 1987.

[7] R. Byrd, E. Eskow, and R. Schnabel, *A new large-scale global optimization method and its application to Lennard-Jones problems*, Tech. Report CU-CS-630-92, University of Colorado at Boulder, 1992.

[8] R. Byrd, E. Eskow, R. Schnabel, and S. Smith, *Parallel global optimization: Numerical methods, dynamic scheduling methods, and applications to molecular configuration*, Tech. Report CU-CS-553-91, University of Colorado at Boulder, 1991.

[9] R.H. Byrd, J. Nocedal, R.B. Schnabel *Representations of quasi-Newton matrices and their use in limited memory methods*, Math. Prog. (Ser. A), 63:129–156, 1994.

[10] G. Ciccotti, M. Ferrario, and J.-P. Ryckaert *Molecular dynamics of rigid systems in cartesian coordinates: a general formulation*, Molec. Phys., 47:1253–1264, 1982.

[11] D. Cvijovic and J. Klinowski, *Taboo search: An approach to the multiple minima problem*, Science, 267 (1995), p. 664.

[12] J. Dennis and V. Torczon, *Direct search methods on parallel machines*, SIAM J. Optimization, 1 (1991), pp. 448–474.

[13] J.J. Dongarra, J.R. Bunch, C.B. Moler, and G.W. Stewart *LINPACK User's Guide* Philadelphia: SIAM, 1979.

[14] R. Fletcher *Practical Methods of Optimization* Second ed. Chichester, UK: Wiley & Sons, 1990.

[15] P. E. Gill, W. Murray, and M. H. Wright *Practical Optimization* London: Academic Press – Harcourt, 1981.

[16] N. Gō and H.A. Scheraga *Ring closure and local conformational deformations of chain molecules*, Macromolecules, 3:178–187, 1970.

[17] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.

[18] G. H. Golub and C. F. Van Loan. *Matrix Computations* Second ed. Baltimore: Johns Hopkins UP, 1991.

[19] M.R. Hoare, *Structure and dynamics of simple microclusters*, Adv. Chem. Phys., 40:49–135, 1979.

[20] R. Hooke and T. Jeeves, *Direct search solution of numerical and statistical problems*, J. Assoc. Comp. Mach., 8 (1961), pp. 212–229.

[21] A. Howard and P. Kollman, *An analysis of current methodologies for conformational searching of complex molecules*, J. Med. Chem., 31 (1988), pp. 1669–1675.

[22] R. Judson, D. Barsky, T. Faulkner, D. McGarrah, C. Melius, J. Meza, E. Mori, and T. Plantenga, *CCEMD - Center for Computational Engineering molecular dynamics: Theory and users' guide, version 2.2*, Tech. Report SAND95-8258, Sandia National Laboratories, 1995.

[23] R. Judson, M. Colvin, J. Meza, A. Huffer, and D. Gutierrez, *Do intelligent configuration search techniques outperform random search for large molecules?*, International Journal of Quantum Chemistry, 44 (1992), pp. 277–290.

[24] R. Judson, E. Jaeger, A. Treasurywala, and M. Peterson, *Conformational searching methods for small molecules II: A genetic algorithm approach*, J.Comp.Chem., 14 (1993), p. 1407.

[25] J. Kostrowicki, L. Piela, B. Cherayil, and H. Scheraga, *Performance of the diffusion equation method in searches for optimum structures of clusters of Lennard-Jones atoms*, J.Phys.Chem., 95 (1991), p. 4113.

[26] M. Lalee, J. Nocedal, and T. Plantenga *On the implementation of an algorithm for large-scale equality constrained optimization*, Submitted to SIAM J. Optimization, 1993.

[27] S.M. Le Grand and K.M. Merz Jr. *The application of the genetic algorithm to the minimization of potential energy functions*, Journal of Global Optimization, Vol. 3.1 (1993), pp. 49-66.

[28] D.C. Liu and J. Nocedal *On the limited memory BFGS method for large scale optimization*, Math. Prog. (Ser. B), 45:503–525, 1989.

[29] R. Maier, J. Rosen, and G. Xue, *Discrete-continuous algorithm for molecular energy mini-*

mization, Tech. Report 92-031, AHPCRC, 1992.

[30] C. Maranas and C. Floudas, *A deterministic global optimization approach for molecular structure determination*, J.Chem.Phys., 100 (1994), p. 1247.

[31] A.K. Mazur, V.E. Dorofeev, and R.A. Abagyan *Derivation and testing of explicit equations of motion for polymers described by internal coordinates*, J. Comput. Phys., 92:261–272, 1991.

[32] J. Meza, R. Judson, T. Faulkner, and A. Treasurywala, *A comparison of a direct search method and a genetic algorithm for conformational searching*, Tech. Report SAND95-8225, Sandia National Laboratories, 1995. To appear in the J. Comp. Chem., 1996.

[33] J. Meza and M. Martinez, *Direct search methods for the molecular conformation problem*, Journal of Computational Chemistry, 15 (1994), pp. 627–632.

[34] F.A. Momany, R.F. McGuire, A.W. Burgess, and H.A. Scheraga, *Geometric parameters, partial atomic charges, nonbonded interactions, hydrogen bond interactions, and intrinsic torsional potentials for the naturally occurring amino acids*, J. Phys. Chem., 79:2361–2381, 1975.

[35] J. Nelder and R. Mead, *A simplex method for function minimization*, Comput. J., 7 (1965), pp. 308–313.

[36] E.O. Omojokun, *Trust region algorithms for optimization with nonlinear equality and inequality constraints*, Diss. Dept. of Computer Science, University of Colorado, 1989.

[37] P.M. Pardalos, D. Shalloway, and G. Xue (Editors) *Optimization Methods for Computing Global Minima of Nonconvex Potential Energy Functions*, Journal of Global Optimization, Vol. 4.2 (1994), pp. 117-133.

[38] P.M. Pardalos, D. Shalloway, and G. Xue (Editors) *Global Minimization of Nonconvex Energy Functions: Molecular Conformation and Protein Folding*, DIMACS Series, Vol. 23, American Mathematical Society (1996).

[39] T.D. Plantenga *Large-scale nonlinear constrained optimization using trust regions*, Diss. Dept. of Electrical Engineering and Computer Science, Northwestern University, 1994.

[40] T. Plantenga and R. Judson, *Energy minimization along dihedrals in cartesian coordinates using constrained optimization*, Tech. Report SAND95-8724, Sandia National Laboratories, 1995.

[41] QUANTA/CHARMM, Molecular Simulations, Inc. (Waltham MA, 1993). The results published were generated in part using the program QUANTA. This program was developed by Molecular Simulations, Inc.

[42] M. Saunders, *Stochastic search for the conformations of bicyclic hydrocarbons*, J.Comp.Chem., 10 (1989), p. 203.

[43] M. Saunders, K. Houk, Y.-D. Wu, W. C. Still, M. Lipton, G. Chang, and W. C. Guida, *Conformations of cycloheptadecane. A comparison of methods for conformational searching*, J. Am. Chem. Soc., 112 (1990), pp. 1419–1427.

[44] S. Sunada and N. Gō *Small-amplitude protein conformational dynamics: second-order analytic relation between Cartesian coordinates and dihedral angles*, J. Comp. Chem., 16:328–336, 1995.

[45] S. Wilson and W. Cui, *Applications of simulated annealing to peptides*, Biopolymers, 29 (1990), pp. 225–235.

[46] G. Xue, *Improvement on the Northby algorithm for molecular conformation: Better solutions*, Tech. Report 92-055, University of Minnesota, 1992.

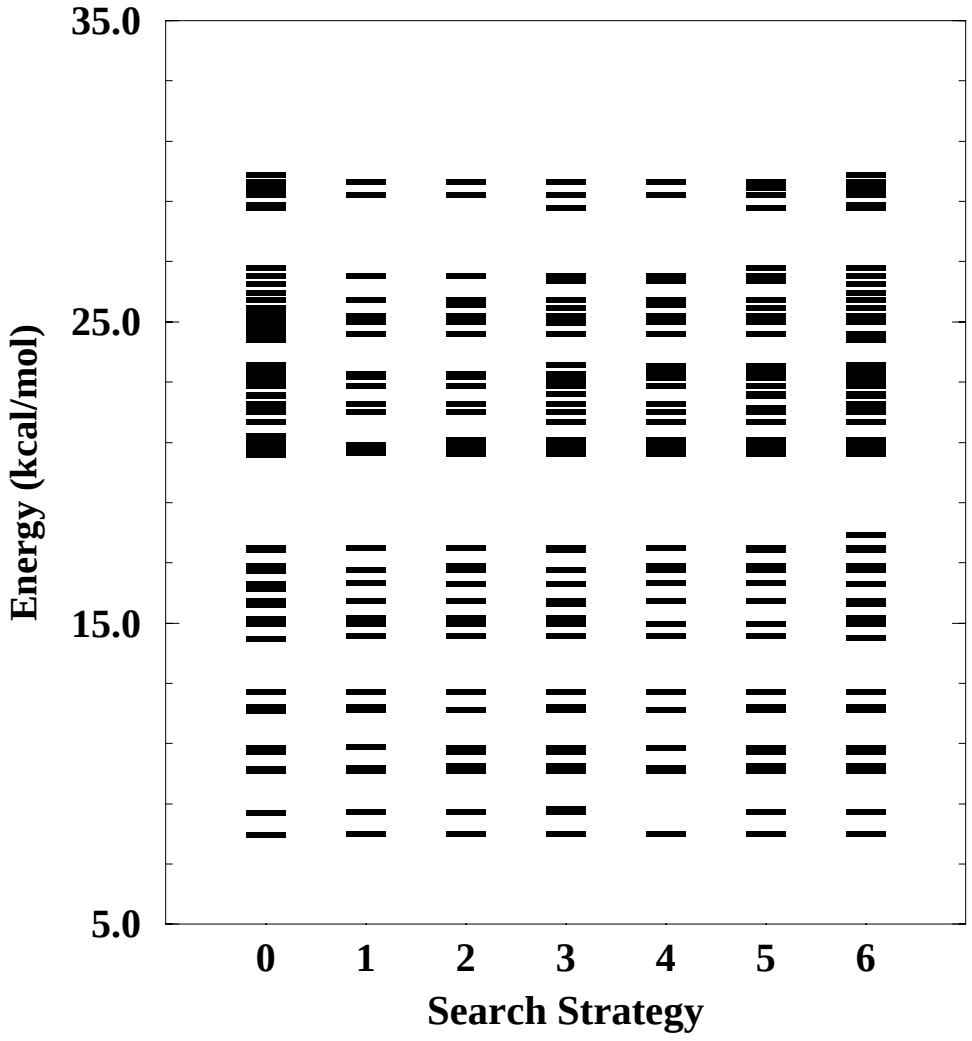

FIG. 3. *Energy spectrum for minimizing Thr-Ala in torsion space. Each mark shows the energy of a unique local minimum. The column numbers correspond to global search strategies.*

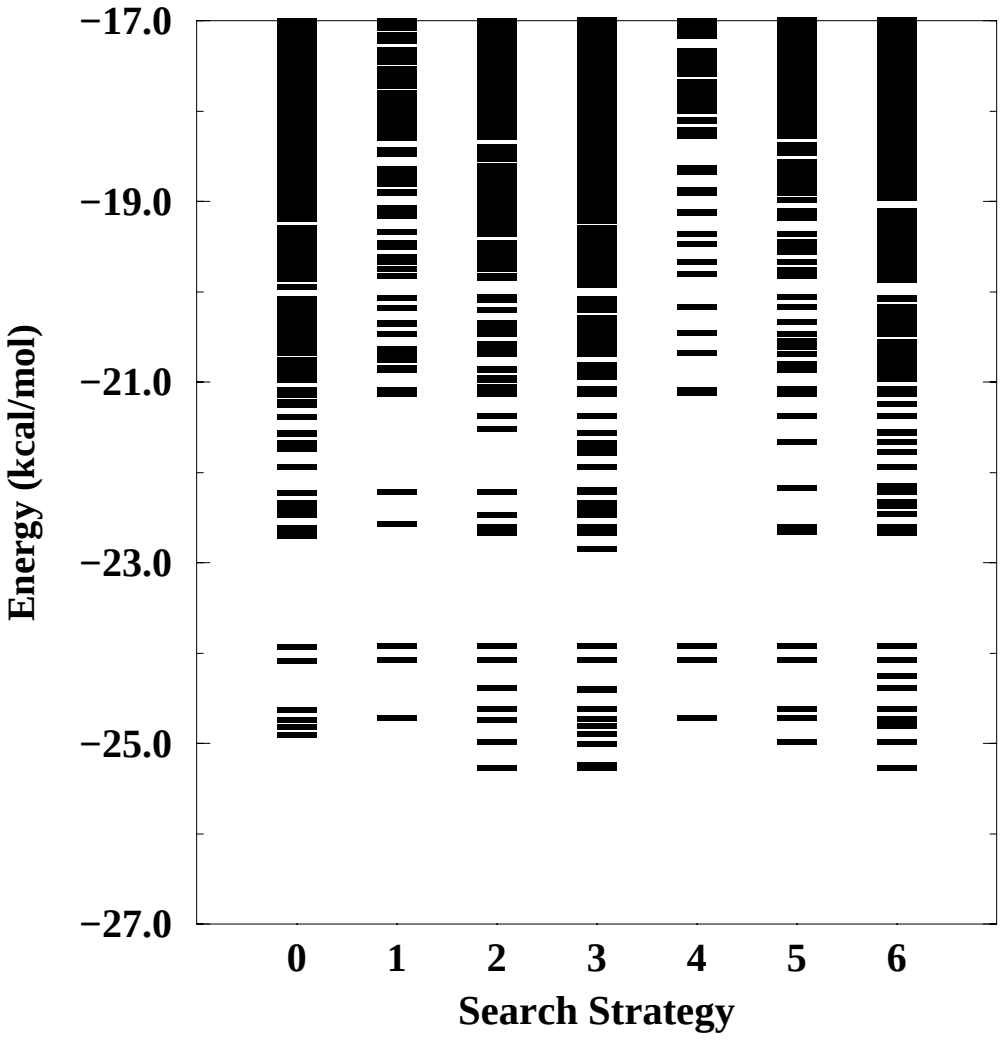

FIG. 4. *Low energy spectrum for minimizing Thr-Ala-Leu in torsion space. Each mark shows the energy of a unique local minimum. The column numbers correspond to global search strategies.*